Amendment to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A processing block comprising:

a storage sub-block;

an execution sub-block to execute instructions; and

a thread management sub-block coupled to the storage and execution sub-blocks, and equipped to store and maintain a thread switching structure in the storage sub-block to facilitate interleaving execution of a plurality of threads of instructions by the execution sub-block, with the thread switching structure including a current thread identifier identifying one of the plurality of threads as a current thread being currently executed by the execution sub-block, and a thread array of thread entries, one per thread, correspondingly describing the plurality of threads, each thread entry being created and added to the thread array by the thread management sub-block as part of the execution of a create thread instruction of a thread to spawn execution of another thread.

2. (Original) The processing block of claim 1, wherein each thread entry comprises a thread program counter to identify an instruction of the corresponding described thread as a current instruction to be executed, when the corresponding described thread is being executed.

3. (Original) The processing block of claim 1, wherein each thread entry comprises an activeness indicator indicating whether the corresponding described thread is currently in an active state or an inactive state, where the corresponding described thread is to be included among the threads to be interleavingly executed by the execution sub-block, while the thread is in the active state, and not included, while the thread is in the inactive state.

4. (Original) The processing block of claim 3, wherein the thread management sub-block is equipped to reset the activeness indicator of a thread from the active state to the inactive state, as

part of the execution of a thread termination instruction of a thread terminating its own execution.

5. (Original) The processing block of claim 1, wherein each thread entry comprises thread dependency information describing at least a plurality of registers of an external set of registers, on which the corresponding described thread depends.

6. (Original) The processing block of claim 1, wherein the processing block further comprises an interface to couple the processing block to an external set of registers.

7. (Original) The processing block of claim 1, wherein the thread management sub-block is further equipped to select a non-current one of the plurality of threads to be the new current thread to be executed, updating the current thread identifier and switching execution to a first instruction of the new current thread accordingly, as part of the execution of a thread switching instruction of a thread instructing the execution sub-block to switch execution to another thread.

8. (Original) The processing block of claim 7, wherein the execution sub-block is equipped to select the next current thread on a selected one of a round-robin basis, a fixed priority basis, and a rotating priority basis.

9. (Original) The processing block of claim 1, wherein the processing block further comprises an input/output interface configurable to be a selected one of an input interface and an output interface to particularize the processing block as a selected one of an input processing block and an output processing block of a signal processing macroblock.

10. (Original) The processing sub-block of claim 1, wherein the processing sub-block further comprises another storage sub-block coupled to the execution sub-block, to store instructions of the threads.

11. (Currently amended) In a processing block, an execution method, comprising:
fetching a first instruction of a first thread of instructions; and

executing the first instruction, and as part of the execution of the first instruction, adding a first thread entry in a thread array of a thread switching structure, if the first instruction is a create thread instruction spawning interleaved execution of a second thread of instructions with execution of other threads, the thread switching structure being disposed and maintained within ~~the processing sub-block~~a storage sub-block by a thread management sub-block to facilitate interleaved execution of threads of instructions by the processing sub-block, and the first thread entry describing the second thread.

12. (Original) The method of claim 11, wherein the first thread entry comprises a thread program counter to identify an instruction of the second thread as a current instruction to be executed, when the second thread is being executed.

13. (Original) The method of claim 11, wherein the first thread entry comprises an activeness indicator indicating whether the second thread is currently in an active state or an inactive state, where the second thread is to be included among the threads to be interleavingly executed, while the second thread is in the active state, and not included, while the second thread is in the inactive state.

14. (Original) The method of claim 11, wherein the first thread entry comprises thread dependency information describing at least a plurality of registers of an external set of registers, on which the second thread depends.

15. (Original) The method of claim 11, wherein said execution of the first instruction includes resetting an activeness indicator of a second thread entry of the thread array from indicating an active state to indicating an inactive state instead, if the first instruction is a thread termination instruction terminating execution of the first thread, the second thread entry being corresponding to the first thread.

16. (Original) The method of claim 11, wherein said execution of the first instruction includes selecting the second thread from among a plurality of active threads, updating a current thread identifier to identify the second thread as a new current thread to be executed, replacing

the first thread, and transferring execution to an instruction of the second thread instead, if the first instruction is a thread execution switching instruction, and execution of the second thread has previously been spawned.

17. (Original) The method of claim 16, wherein said selection of the second thread from among a plurality of active threads comprises selecting the second thread in a selected one of a round-robin basis, a fixed priority basis, and a rotating priority basis.

18. (Currently amended) A signal processing macroblock comprising:

a set of registers; and

at least a selected one of

an input processing block coupled to the set of registers, including an input interface, execution and thread management facilities equipped to support interleaved execution of multiple threads of instructions, the thread management facilities including a thread management sub-block equipped to maintain a thread switching structure and create and add thread entries corresponding to the multiple threads of instructions to a thread array within the thread switching structure; and

an output processing block coupled to the set of registers, including an output interface, execution and thread management facilities equipped to support interleaved execution of multiple threads of instructions, the thread management facilities including a thread management sub-block equipped to maintain a thread switching structure and create and add thread entries corresponding to the multiple threads of instructions to a thread array within the thread switching structure.

19. (Original) The signal processing macroblock of claim 18, wherein the signal processing macroblock further comprises a computation block coupled to the set of registers, including execution and thread management facilitates equipped to support interleaved execution of multiple threads of instructions, including instructions performing mathematical operations.

20. (Original) The signal processing macroblock of claim 19, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, includes a storage sub-block to store a thread switching structure that includes a current thread identifier identifying one of the multiple threads as a current thread to be executed, and a thread array including thread entries describing corresponding ones of the multiple threads.

21. (Original) The signal processing macroblock of claim 20, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to create a thread entry in the thread array for a thread as part of the execution of a create thread instruction spawning interleaved execution of the thread.

22. (Original) The signal processing macroblock of claim 20, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to reset an activeness indicator of a thread entry in the thread array for a thread from indicating an active state to indicating an inactive state as part of the execution of a thread termination instruction terminating execution of the thread.

23. (Original) The signal processing macroblock of claim 20, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to select a thread as a new current thread to be executed, updating a current thread identifier of the thread switching structure to identify the selected thread, and switching to execute an instruction of the selected thread, as part of the execution of a thread execution switching instruction.

24. (Currently amended) A media processor comprising:

a direct memory access unit to access media data;

a plurality of signal processing units coupled to the direct memory access unit to process the accessed media data, at least a first of which signal processing units comprising

a set of registers,

an input processing block coupled to the set of registers, including an input interface, execution and thread management facilities equipped to support interleaved execution of multiple threads of instructions, the thread management facilities including a thread management sub-block equipped to maintain a thread switching structure and create and add thread entries corresponding to the multiple threads of instructions to a thread array within the thread switching structure; and

an output processing block coupled to the set of registers, including an output interface, execution and thread management facilities equipped to support interleaved execution of multiple threads of instructions, the thread management facilities including a thread management sub-block equipped to maintain a thread switching structure and create and add thread entries corresponding to the multiple threads of instructions to a thread array within the thread switching structure.

25. (Original) The media processor of claim 24, wherein the first signal processing unit further comprises a computation block coupled to the set of registers, including execution and thread management facilitates equipped to support interleaved execution of multiple threads of instructions, including instructions performing mathematical operations.

26. (Original) The media processor of claim 25, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, includes a storage sub-block to store a thread switching structure that includes a current thread identifier identifying one of the multiple threads as a current thread to be executed, and a thread array including thread entries describing corresponding ones of the multiple threads.

27. (Original) The media processor of claim 25, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to create a thread entry in the thread array for a thread as part of the execution of a create thread instruction spawning interleaved execution of the thread.

28. (Original) The media processor of claim 25, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to reset an activeness indicator of a thread entry in the thread array for a thread from indicating an active state to indicating an inactive state as part of the execution of a thread termination instruction terminating execution of the thread.

29. (Original) The media processor of claim 25, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to select a thread as a new current thread to be executed, updating a current thread identifier of the thread switching structure to identify the selected thread, and switching to execute an instruction of the selected thread, as part of the execution of a thread execution switching instruction.

30. (Currently amended) A system comprising:

a host processor;

first memory coupled to the host processor;

second memory;

a media processor coupled to the second memory and the host processor, the media processor having at least

a direct memory access unit to access media data, and

a plurality of signal processing units coupled to the direct memory access unit to process the accessed media, at least a first of which signal processing units comprising

a set of registers,

an input processing block coupled to the set of registers, including an input interface, execution and thread management facilities equipped to support interleaved execution of multiple threads of instructions the thread management facilities including a thread management sub-block equipped to maintain a thread switching structure, and create and add thread entries corresponding to the multiple threads of instructions to a thread array within the thread switching structure;, and

an output processing block coupled to the set of registers, including an output interface, execution and thread management facilities equipped to support interleaved execution of multiple threads of instructions, the thread management facilities including a thread management sub-block equipped to maintain a thread switching structure and create and add thread entries corresponding to the multiple threads of instructions to a thread array within the thread switching structure;.

31. (Original) The system of claim 30, wherein the first signal processing unit of the media processor further comprises a computation block coupled to the set of registers, including execution and thread management facilitates equipped to support interleaved execution of multiple threads of instructions, including instructions performing mathematical operations.

32. (Original) The system of claim 31, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, includes a storage sub-block to store a thread switching structure that includes a current thread identifier identifying one of the multiple threads as a current thread to be executed, and a thread array including thread entries describing corresponding ones of the multiple threads.

33. (Original) The system of claim 31, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to create a thread entry in the thread array for a thread as part of the execution of a create thread instruction spawning interleaved execution of the thread.

34. (Original) The system of claim 31, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to reset an activeness indicator of a thread entry in the thread array for a thread from indicating an active state to indicating an inactive state as part of the execution of a thread termination instruction terminating execution of the thread.

35. (Original) The system of claim 31, wherein at least a selected one of the facilities of the input processing block, the output processing block and the computation block, equipped to support interleaved execution of multiple threads, further includes an execution sub-block equipped to select a thread as a new current thread to be executed, updating a current thread identifier of the thread switching structure to identify the selected thread, and switching to execute an instruction of the selected thread, as part of the execution of a thread execution switching instruction.

36. (Original) The system of claim 30, wherein the system is a selected one of a server, a palm sized personal digital assistant, a wireless mobile phone, a set-top box, an entertainment control console, a video recorder, or a video player.